APPENDIX C

Hands-on Lab Exercises

Some jobs require that you be able to configure routers and switches, but others do not. For instance, many people who sell Cisco products for Cisco or Cisco Channel Partners have never configured a router or switch. However, those same people might know a lot more about other things, like the Cisco product line and what the latest products are. Simply put, some jobs require different skills and knowledge.

Cisco created CCNA as part of an overall plan to assess and verify the skill sets of the various Cisco Channel Partners. The CCNA certification's role was to prove the basic proficiency of a Channel Partner employee in network installation and support. Because CCNA focuses on network installation and support, Cisco wants CCNAs to be able to configure routers and switches.

In order to better test people on whether they have hands-on skills, Cisco includes a practical component on the CCNA exams. The exam engine will simulate routers and switches. Therefore, this chapter is designed to help you practice your hands-on skills.

Is it a good thing that Cisco is adding simulated labs to CCNA? Absolutely! If you have made it this far in the book, you definitely want to learn this stuff, not just pass a test. And the more Cisco can make the exams like a real implementation, asking you to apply the knowledge you have gained rather than just spew forth memorized answers, the more valuable the CCNA certification becomes. Wouldn't you rather configure a small IP network than memorize that pressing Esc-B backs up a single word in the command line? With more hands-on skills on the exam, there will be less time for trivial questions.

So how should you prepare? In a word, practice! Get some routers and switches and do the lab exercises in this chapter—even if you think you understand it all. You can use NetSim network simulator on the accompanying CD to do these labs. Also, use the exam engine on the CD to perform all the simulator-based practice questions on the CD. Build some "muscle memory" for implementing simple networks. Do you think Michael Jordan figured out how to shoot a jump shot when he was 12 and then quit shooting them? No, he practiced them a lot, so now his muscles remember what to do, and he doesn't think about how to shoot a jump shot every time he does it. So, the more you practice configuring routers and switches, the easier it will be to breeze through the exam instead of being hesitant about taking the exam with the new practical component.

Options for Gaining Hands-on Skills

When I sat down to write this chapter, I thought about the different options available when you want to develop hands-on skills. Each option has advantages and disadvantages:

- **Borrowing gear from your company's test lab**—If you already work in a networking job, chances are you can scrounge around and find some gear to use. Of course, getting a combination of gear that matches the examples in the books you're using might be a little more challenging, and collecting the gear each time you have a few hours to study might be a hassle. However, it's still one of the best ways to get hands-on skills. One key is to have some direction as to what to do with the gear after you get it. You can use NetSim network simulator on the accompanying CD to do these labs. Also, use the exam engine on the CD to perform all the simulator-based practice questions on the CD.
- **Buying some gear on the Internet**—You can always buy your own equipment if you do not have access to it at your job. It takes money, but you can always sell the gear when you're done, as long as it isn't broken or obsolete. Collecting a variety of gear to match a book's different examples and scenarios might be difficult, because most books' examples (this one included) are not written to minimize the amount of equipment needed to build the lab. The lab exercises in this chapter use the minimum amount of equipment needed to let you learn the necessary information, hopefully keeping the price down for you.
- **Leasing the gear**—Some companies will lease you a CCNA lab, but you still have the problem of collecting a variety of gear to match the different examples and scenarios in a book.
- **Simulators**—A special version of Boson's NetSim network simulator is included on the CD that comes with this book. This special version allows you to perform the labs in this appendix, as well as some of the other labs and scenarios—see Appendix C in the book for more details. Although simulators cannot teach everything, they are generally adequate for what you need to learn for CCNA.
- **Lab rentals (e-labs)**—Finally, several companies will rent lab time on lab pods accessible from the Internet. These typically can be rented by the hour or by the lab exercise, either for specific lab exercises or to do any labs you would like to perform. The labs in this book should work with some of the lab rental offerings.

About the Labs in This Chapter

The labs in this appendix simply provide practice for the configuration and EXEC commands covered in some of the chapters in the book. These labs assume that you know how to get around the user interface of the router and switch. These practice labs do not tell you exactly what commands and options to type at every step. Instead, they are designed to make you think. The end of this appendix has figures for all the labs and screen captures of worked out lab answers, in case you need help.

Equipment List

You will of course need some equipment in order to do the labs. Because different people might buy different equipment, this list describes the gear generically:

- **Routers (you need two)**—Each router needs one Ethernet and one Serial interface. The serial interface can be a synchronous interface or an async/sync interface. These are priced around $200 to $300 on eBay. If the Ethernet interfaces have an AUI interface, as is the case with the typically cheapest Cisco routers in the used marketplace, you will need AUI transceivers for each router.
- **Router as Frame Relay switch (you need one)**—In all the examples in this book, the Frame Relay network was created using one or more Cisco routers configured to act as a Frame Relay switch. A sample configuration is included in Lab 3 in this chapter. Any Cisco router with two serial interfaces, along with IP IOS, will work. You need this additional router only for Lab 3.
- **Ethernet hubs (you need two) or switch (you need one)**—The routers should be cabled to a separate hub, or cabled to the same switch, with each switch port being configured in a different VLAN. Hubs are really cheap today; any switch that supports VLANs can be used as well.
- **LAN cables (you need two)**—Two Category 5 straight-through cables attach the routers to the switch or hub. If your two routers use an AUI interface, you need AUI transceivers as well.
- **Serial cables**—You need one DCE cable and one DTE cable, which will be connected to create the serial link between R1 and R2. The connectors are dependent on the connectors on the routers you buy.
- **Serial cables (for Frame Relay)**—You need an additional DCE cable and DTE cable pair, which will be connected, between each router and the router acting as a Frame Relay switch. Depending on the type of interfaces on the Frame Relay router, you might not be able to use the ones you used to create the point-to-point link between R1 and R2.
- **Console kit (you need one)**—You need to access the console of the routers and switches. A console kit contains the correct cable and connector.

For example, you could obtain the following equipment. This list contains currently–popularly–available items on eBay:

- Three 2501 routers
- One VLAN capacble switch or two hubs
- Two Ethernet AUI transceivers
- Two DB60-V.35 DTE cables (part number CAB-v35MT=)
- Two DB60-V.35 DCE cables (part number CAB-v35FC=)
- Two 6-foot Category 5 cables (straight-through)
- One console cable/connector

Using NetSim

You can use the NetSim network simulation software on the CD to perform the labs in this chapter. Appendix C in the book contains some hints and tips for loading NetSim, as well as listing some caveats when using NetSim to do labs from the book. Please refer to Appendix C in the (printed) book for more information.

List of Labs

Table C-1 describes the labs in this chapter. Please refer to Appendix C in the book for a list of all labs that can be done using the NetSim simulator.

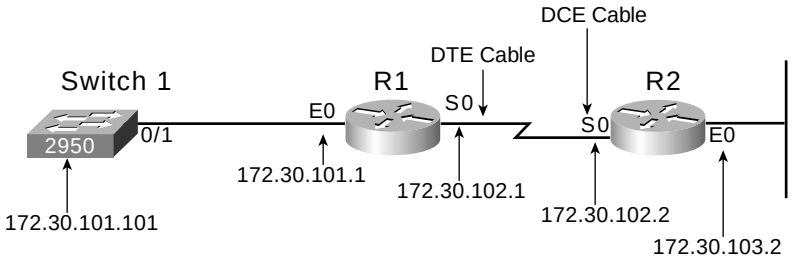
Table C-1 *Lab Descriptions*

Lab	Title	Description
1	IP Routing Configuration	This lab has you configure static IP routes, as well as the RIP and IGRP routing protocols.
2	IP Access List Configuration	This lab has you configure two different extended IP access lists and one named extended IP access list.
3	WAN Configuration	This lab takes you through PPP configuration, as well as several Frame Relay configuration options.

Lab 1: IP Routing Configuration

This lab assumes that you know how to configure IP addresses and know how to navigate the user interface on the routers. This lab assumes you are using a lab network that matches Figure C-1.

Figure C-1 *Network Topology for Lab 1*



Lab 1: Objectives

When finished with this lab, you will be able to do the following:

- Configure static IP routes
- Configure RIP
- Configure IGRP
- Verify the contents of the IP routing table

Lab 1: Step-by-Step Instructions

Step 1 This lab begins with a working network based on the last lab in the *INTRO Exam Certification Guide*, CD appendix C. You can refer to those instructions, or you can simply configure the lab based on the figure. Configure both routers with the IP addresses shown, set the clock rate on R2's Serial0 using the **clock rate 56000** command, use the **no shutdown** command on each interface. Do not configure a routing protocol yet—you will do that in this lab!

Step 2 You should be able to ping 172.30.102.2, R2's serial IP address, but not 172.30.103.2, R2's Ethernet IP address. The ping to 172.30.103.2 does not work, because R1 does not have a route to 172.30.103.0/24, the subnet in which 172.30.103.2 resides.

Step 3 Verify the contents of the routing table on R1. What command did you use? How many routes are in the routing table?

Step 4 On R1, configure a static route pointing to subnet 172.30.103.0.

Step 5 On R1, ping 172.30.103.2. Does it work now? (It should!) If it doesn't, work on your static route configuration until it does.

Step 6 From R2, try to ping R1's Ethernet IP address, 172.30.101.1. Does it work? (It should not.)

Step 7 Add a static route on R2 that will make the ping to 172.30.101.1 work.

Step 8 How many routes are in R2's routing table?

Step 9 Remove the static route commands from both R1 and R2. How many routes are in each router's routing table now?

Step 10 Configure RIP on both R1 and R2.

Step 11 From R1, try pinging R2's Ethernet IP address. Similarly, ping R1's Ethernet IP address from R2. Does it work?

Step 12 What is the administrative distance associated with the routes learned by RIP? Record the command you use to discover this information.

Step 13 Configure IGRP on R1 and R2. How many routes are in each routing table now?

Step 14 What are the sources of the routing information? Static? Connected? RIP? IGRP? EIGRP?

Step 15 Turn on a debug that shows IGRP routing updates. How often do they occur? (NetSim does not currently support IGRP debugs. Refer to the answer to this lab, later in the appendix, to see a sample of the debugs.)

Step 16 Turn on a debug that shows RIP routing updates. How often do they occur?

Step 17 Does it appear that RIP is still sending and receiving routing updates?

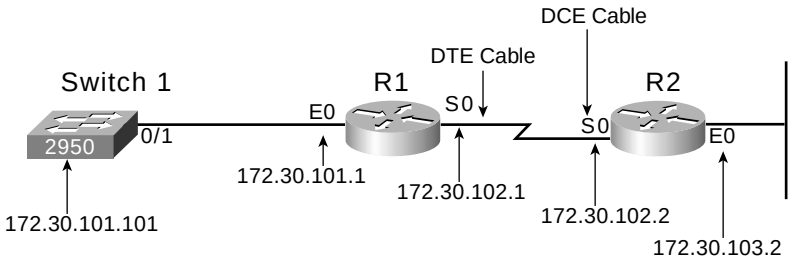
Step 18 Turn off RIP on both routers. Are all the routes still in the routing table?

Step 19 Turn off your debugs, and save your configurations on both routers. (Those of you using NetSim, you must use the **undebug all** command to disable the debugs.)

Lab 2: IP Access List Configuration

This lab assumes that you have completed Lab 1 in this appendix, by the end of which you had a working lab network using IGRP. This lab assumes that you know how to get around the user interface on the routers. This lab assumes you are using a lab network that matches that in Figure C-2.

Figure C-2 *Network Topology for Lab 2*



Lab 2: Objectives

When finished with this lab, you will be able to do the following:

- Configure extended IP access lists
- Configure named extended IP access lists

Lab 2: Step-by-Step Instructions

Step 1 Log in to R1.

Step 2 Verify that you can ping 172.30.103.2, R2's Ethernet0 IP address. Likewise, verify that you can Telnet to that same address.

Step 3 After you have Telnetted to R2, configure the Web server feature on R2 using the **ip http server** configuration command. Save your configuration on R2.

Step 4 Suspend/quit back to R1.

Step 5 From R1, you can use the **telnet** command to test whether the Web server on R2 is responding, using a trick. From R1, use the command **telnet 172.30.103.2 80**. This command uses Telnet, but it tries to Telnet to destination port 80—the port used by the Web server. You should get no error messages if this works correctly. Then, type ?. You see some messages from the Web server on R2 about HTTP 400 bad request, as shown in Example C-1. (For NetSim users, this particular feature is not supported—so ignore the configuration to filter web traffic.)

Example C-1 *Trick to Test Web Traffic Using Just Routers*

```
R1#telnet 172.30.103.2 80
Trying 172.30.103.2, 80 ... Open
?
HTTP/1.0 400 Bad Request
Date: Mon, 01 Mar 1993 03:08:27 UTC
Content-type: text/html
Expires: Thu, 16 Feb 1989 00:00:00 GMT

<H1>400 Bad Request</H1>



[Connection to 172.30.103.2 closed by foreign host]
```

Step 6 So far, you should have confirmed that ping, Telnet, and Web traffic to 172.30.103.2 work.

Step 7 Create an access list on R2, and enable it for traffic entering R2's S0 interface. In this access list, permit ICMP echoes to the Ethernet subnet off R2, permit Telnet to all IP addresses in that subnet except R2's Ethernet IP address, and allow Web traffic to any Web servers in that same subnet. Also, permit Telnet into 172.30.102.2 so that you can still Telnet into R2 to do your testing.

Step 8 After configuring the access list, from R1, test. A **ping** from R1 to 172.30.103.2 should work, a **telnet** to the same address should not, and the **telnet 172.30.103.2 80** trick should still work.

Step 9 Disable the access list on R2 so that all traffic is now allowed.

Step 10 On R2, create a named access list that achieves the same goal.

Step 11 Test again from R1. A **ping** from R1 to 172.30.103.2 should work, a **telnet** to the same address should not, and the **telnet 172.30.103.2 80** trick should still work.

Step 12 Disable the named access list on R2 so that all traffic is now allowed.

Step 13 Create an access list on R1, enabled for input traffic in R1's S0 interface, that achieves the same function as the first ACL.

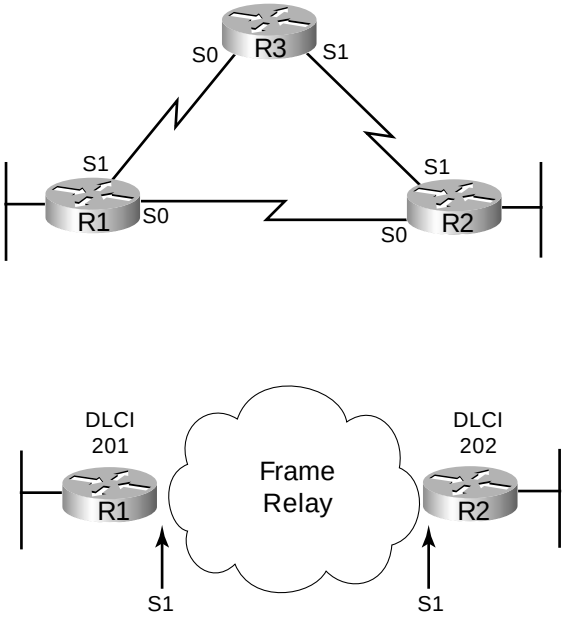
Step 14 Test again from R1. A **ping** from R1 to 172.30.103.2 should work, a **telnet** to the same address should not, and the **telnet 172.30.103.2 80** trick should still work.

Step 15 When finished, disable the access list on R1.

Lab 3: WAN Configuration

This lab assumes that you have completed Labs 2 and 3, in which you configured IP addresses and routing protocols. This lab assumes that you know how to get around the user interface on the routers. This lab requires that you use three routers, with one configured as a Frame Relay switch. Figure C-3 outlines the physical and logical diagrams for the network.

Figure C-3 *Network Topology for Lab 3*



Lab 3: Objectives

When finished with this lab, you will be able to do the following:

- Configure PPP
- Configure Frame Relay

Lab 3: Step-by-Step Instructions

Step 1 The first several steps of this lab use the direct serial link between R1 and R2 that you were using in earlier labs. So, log in to R1 and R2 and ensure that all ACLs from the previous lab have been disabled.

Step 2 On R1, change the encapsulation type on Serial0 to PPP encapsulation. From R1, ping R2's serial IP address. Does it still work?

Step 3 On R2, change the encapsulation type on Serial0 to PPP. Go back to R1. Can R1 ping R2's serial IP address now?

Step 4 Examine the IP routing table. On R1, is there a route to 172.30.102.2? Is there a similar route on R2?

Step 5 On R1, enable PPP debugs using the **debug PPP negotiation** command. (NetSim does not yet support debugs for PPP. Refer to the answer to this lab, later in the appenidix, to see a sample of the debugs.)

Step 6 On R1, shut down the serial link using the **shutdown** command, wait 5 seconds, and bring it back up with the **no shutdown** command. Get out of configuration mode as soon as you issue the **no shutdown** command.

Step 7 Debug messages should appear, giving you some insight into the PPP negotiation that occurs when PPP initializes. Notice the message that shows PPP on R1 learning about R2's IP address, and R1 telling R2 its own IP address.

Step 8 Shut down the Serial0 interface on both R1 and R2.

Step 9 Log in to the console of R3, and copy the configuration shown in Example C-2 into R3. (NetSim may have already created the Frame Relay cloud.)

Example C-2 *Frame Relay Switch Configuration*

```
hostname FRswitch
!
frame-relay switching
!
interface Serial0
 no ip address
 encapsulation frame-relay
 clockrate 56000
 no shutdown
 frame-relay intf-type dce
 frame-relay route 202 interface Serial1 201
!
interface Serial1
 no ip address
 encapsulation frame-relay
 clockrate 56000
 no shutdown
 frame-relay intf-type dce
 frame-relay route 201 interface Serial0 202
```

Step 10 Cable the serial cables between R1 and R3 and between R2 and R3 based on Figure C-3. Make sure that the DCE cable ends are plugged into the R3. Shutdown the serial link directly between R1 and R2.

Step 11 Configure Frame Relay between R1 and R2 with the following settings: use LMI autosense (the default), use Cisco encapsulation (the default), allow Inverse ARP to work (the default), and do not use subinterfaces.

Use a new set of IP addresses instead of the point-to-point link used in the other labs. Use 172.30.104.0/24, using 172.30.104.1 on R1 and 172.30.104.2 on R2.

Step 12 From R1, test the network by pinging R2's IP addresses. Likewise, ping R1 from R2. Do not proceed past this step until it is working. Feel free to refer to the solution listed later in this appendix for an example of a working configuration.

Step 13 From R1, examine the mapping between R2's IP address and the associated DLCI. What is the PVC's DLCI? How long has the associated DLCI been up? Record the commands you use.

Step 14 Disable Inverse ARP on both routers, and then try to ping the routers on the other side of the network. Record the results.

Step 15 On both routers, configure static **frame-relay map** statements to map the next-hop IP address to the correct DLCI value. Record the commands that you use. Verify that it works correctly by pinging R2 from R1 and vice versa.

Step 16 Convert the configuration to use point-to-point subinterfaces on both R1 and R2. Delete the old **frame-relay static map** statements. Can you ping the other routers?

Step 17 Look at the current PVC status and the Frame Relay map information. Is there more or less information than with the earlier configuration?

Step 18 Confirm that R1 can ping R2. Then change R1's Frame Relay encapsulation type to ietf, and try the ping again. Does it work? Why?

Step 19 Change the Frame Relay encapsulation type on R2 to ietf, and try the ping again. Does it work? Why?

Step 20 What LMI type is used by R1? By R2? Record the command you use to discover the LMI type.

Lab Answers

Lab 1: IP Routing Configuration

Example C-3 *Solution to Lab 1*

```
R1>
R1>enable
Password:
R1#ping 172.30.102.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.102.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/36/36 ms
R1#ping 172.30.103.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.103.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
!
! Step 3 next. Only connected routes, because no routing protocols
! are configured yet.
!
R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 2 subnets
C       172.30.102.0 is directly connected, Serial0
C       172.30.101.0 is directly connected, Ethernet0
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip route ?
  A.B.C.D  Destination prefix
  profile  Enable IP routing table profile
  vrf      Configure static route for a VPN Routing/Forwarding instance
```

Example C-3 *Solution to Lab 1 (Continued)*

```
R1(config)#ip route 172.30.103.0 ?
  A.B.C.D  Destination prefix mask

R1(config)#ip route 172.30.103.0 255.255.255.0 ?
  A.B.C.D            Forwarding router's address
  Async              Async interface
  BVI                Bridge-Group Virtual Interface
  CTunnel            CTunnel interface
  Dialer             Dialer interface
  Ethernet           IEEE 802.3
  Lex                Lex interface
  Loopback           Loopback interface
  Multilink          Multilink-group interface
  Null               Null interface
  Serial             Serial
  Tunnel             Tunnel interface
  Vif                PGM Multicast Host interface
  Virtual-Template   Virtual Template interface
  Virtual-TokenRing  Virtual TokenRing

R1(config)#ip route 172.30.103.0 255.255.255.0 serial 0
R1(config)#^Z
!
! IP route to R2's Ethernet subnet, pointing out R1's S0.
!
R1#ping 172.30.103.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.103.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/35/36 ms
R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 3 subnets
C       172.30.102.0 is directly connected, Serial0
S       172.30.103.0 is directly connected, Serial0
C       172.30.101.0 is directly connected, Ethernet0
!
! Routing table in previous command  step 5 completed. Telnetting to
```

continues

Example C-3 *Solution to Lab 1 (Continued)*

```
! R2 next (step 6)
!
R1#r2
Trying R2 (172.30.102.2)... Open


User Access Verification

Password:
R2>enable
Password:
R2#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 2 subnets
C       172.30.102.0 is directly connected, Serial0
C       172.30.103.0 is directly connected, Ethernet0
R2#ping 172.30.102.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.102.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 40/44/64 ms
R2#ping 172.30.101.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.101.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
!
! Previous commands just confirming the ping from R2 to R1's Ethernet
! still does not work.
!
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#ip route 172.30.101.0 255.255.255.0 ?
  A.B.C.D            Forwarding router's address
  Async              Async interface
  BVI                Bridge-Group Virtual Interface
  CTunnel            CTunnel interface
```

Example C-3 *Solution to Lab 1 (Continued)*

```
  Dialer             Dialer interface
  Ethernet           IEEE 802.3
  Lex                Lex interface
  Loopback           Loopback interface
  Multilink          Multilink-group interface
  Null               Null interface
  Serial             Serial
  Tunnel             Tunnel interface
  Vif                PGM Multicast Host interface
  Virtual-Template   Virtual Template interface
  Virtual-TokenRing  Virtual TokenRing

R2(config)#ip route 172.30.101.0 255.255.255.0 172.30.102.1
R2(config)#^Z
R2#ping 172.30.101.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.101.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 40/44/60 ms
R2#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 3 subnets
C       172.30.102.0 is directly connected, Serial0
C       172.30.103.0 is directly connected, Ethernet0
S       172.30.101.0 [1/0] via 172.30.102.1
!
! Finished with step 8. Removing the static routes next.
!
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#no ip route 172.30.101.0 255.255.255.0 172.30.102.1
R2(config)#^Z
R2#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
```

continues

Example C-3 *Solution to Lab 1 (Continued)*

```
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 2 subnets
C       172.30.102.0 is directly connected, Serial0
C       172.30.103.0 is directly connected, Ethernet0
R2#
!
! Still on step 9, just suspending Telnet back to R1.
!
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#no ip route 172.30.103.0 255.255.255.0 serial
R1(config)#^Z
R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 2 subnets
C       172.30.102.0 is directly connected, Serial0
C       172.30.101.0 is directly connected, Ethernet0
R1#
!
! Now to step 10, configuring RIP.
!
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router rip
R1(config-router)#network 172.30.0.0
R1(config-router)#^Z
R1#
!
! Just pressing Enter to resume connection to R2. Continuing step 10.
!
[Resuming connection 1 to r2 ... ]

R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#router rip
```

Example C-3 *Solution to Lab 1 (Continued)*

```
R2(config-router)#network 172.30.0.0
R2(config-router)#^Z
R2#
!
! Suspending back to R1. Testing at step 11 next.
!
R1#ping 172.30.103.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.103.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/36/36 ms
R1#
!
! Just press Enter, resume suspended Telnet, back to R2, to test
! for step 11.
!
[Resuming connection 1 to r2 ... ]

R2#ping 172.30.101.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.101.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 40/44/60 ms
R2#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 3 subnets
C       172.30.102.0 is directly connected, Serial0
C       172.30.103.0 is directly connected, Ethernet0
R       172.30.101.0 [120/1] via 172.30.102.1, 00:00:25, Serial0
R2#
!
! Ctrl-Shift-6, x to suspend back to R1. Step 13 next.
!
R1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router igrp 1
R1(config-router)#network 172.30.0.0
```

continues

Example C-3 *Solution to Lab 1 (Continued)*

```
R1(config-router)#^Z
R1#
[Resuming connection 1 to r2 ... ]

R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#router igrp 1
R2(config-router)#network 172.30.0.0
R2(config-router)#^Z
R2#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 3 subnets
C       172.30.102.0 is directly connected, Serial0
C       172.30.103.0 is directly connected, Ethernet0
I       172.30.101.0 [100/8576] via 172.30.102.1, 00:00:02, Serial0
!
! Note the route learned by RIP is gone, replaced by the IGRP
! learned route. IGRP's admin distance is 100, as compared to
! RIP's 120  so the IGRP learned route is better.
!
R2#debug ip igrp ?
  events        IGRP protocol events
  transactions  IGRP protocol transactions

R1#debug ip igrp events
IGRP event debugging is on
R1#debug ip rip ?
  database  RIP database events
  events    RIP protocol events
  trigger   RIP trigger extension
  <cr>

R1#debug ip rip events
RIP event debugging is on
R1#
00:52:49: RIP: received v1 update from 172.30.102.2 on Serial0
00:52:49: RIP: Update contains 1 routes
00:53:07: RIP: sending v1 update to 255.255.255.255 via Ethernet0 (172.30.101.1)
00:53:07: RIP: Update contains 2 routes
```

Example C-3 *Solution to Lab 1 (Continued)*

```
00:53:07: RIP: Update queued
00:53:07: RIP: sending v1 update to 255.255.255.255 via Serial0 (172.30.102.1)
00:53:07: RIP: Update contains 1 routes
00:53:07: RIP: Update queued
00:53:07: RIP: Update sent via Ethernet0
00:53:07: RIP: Update sent via Serial0
00:53:18: RIP: received v1 update from 172.30.102.2 on Serial0
00:53:18: RIP: Update contains 1 routes
00:53:20: IGRP: sending update to 255.255.255.255 via Ethernet0 (172.30.101.1)
00:53:20: IGRP: Update contains 2 interior, 0 system, and 0 exterior routes.
00:53:20: IGRP: Total routes in update: 2
00:53:20: IGRP: sending update to 255.255.255.255 via Serial0 (172.30.102.1)
00:53:20: IGRP: Update contains 1 interior, 0 system, and 0 exterior routes.
00:53:20: IGRP: Total routes in update: 1
00:53:27: IGRP: received update from 172.30.102.2 on Serial0
00:53:27: IGRP: Update contains 1 interior, 0 system, and 0 exterior routes.
00:53:27: IGRP: Total routes in update: 1
00:53:34: RIP: sending v1 update to 255.255.255.255 via Ethernet0 (172.30.101.1)
00:53:34: RIP: Update contains 2 routes
00:53:34: RIP: Update queued
00:53:34: RIP: sending v1 update to 255.255.255.255 via Serial0 (172.30.102.1)
00:53:34: RIP: Update contains 1 routes
00:53:34: RIP: Update queued
00:53:34: RIP: Update sent via Ethernet0
00:53:34: RIP: Update sent via Serial0
R1#no debug all
All possible debugging has been turned off
!
! The real answer to "how often" is found in the next command.
! It's a lot easier than timing debug messages!!
!
R1#show ip protocol
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 15 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is
  Incoming update filter list for all interfaces is
  Redistributing: rip
  Default version control: send version 1, receive any version
    Interface             Send  Recv  Triggered RIP  Key-chain
    Ethernet0             1     1 2
    Serial0               1     1 2
  Automatic network summarization is in effect
  Routing for Networks:
    172.30.0.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    172.30.102.2         120      00:00:26
```

continues

Example C-3 *Solution to Lab 1 (Continued)*

```
  Distance: (default is 120)

Routing Protocol is "igrp 1"
  Sending updates every 90 seconds, next due in 28 seconds
  Invalid after 270 seconds, hold down 280, flushed after 630
  Outgoing update filter list for all interfaces is
  Incoming update filter list for all interfaces is
 --More
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  IGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  IGRP maximum hopcount 100
  IGRP maximum metric variance 1
  Redistributing: igrp 1
  Routing for Networks:
    172.30.0.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    172.30.102.2         100      00:00:47
  Distance: (default is 100)

R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 3 subnets
C       172.30.102.0 is directly connected, Serial0
I       172.30.103.0 [100/8576] via 172.30.102.2, 00:00:56, Serial0
C       172.30.101.0 is directly connected, Ethernet0
!
! Now on to step 18, turning off RIP.
!
R1#configure t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#no router rip
R1(config)#^Z
R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
```

Example C-3 *Solution to Lab 1 (Continued)*

```
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 3 subnets
C       172.30.102.0 is directly connected, Serial0
I       172.30.103.0 [100/8576] via 172.30.102.2, 00:00:04, Serial0
C       172.30.101.0 is directly connected, Ethernet0
R1#
[Resuming connection 1 to r2 ... ]

R2#configure t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#no router rip
R2(config)#^Z
R2#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 3 subnets
C       172.30.102.0 is directly connected, Serial0
C       172.30.103.0 is directly connected, Ethernet0
I       172.30.101.0 [100/8576] via 172.30.102.1, 00:00:03, Serial0
!
! Last step, save your configs.
!
R2#copy run start
Destination filename [startup-config]?
Building configuration...

R1#copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
R1#
```

Lab 2: IP Access List Configuration

Example C-4 *Solution to Lab 2*

```
R1>
R1>enable
Password:
!
! Step 2 - ping and Telnet to 172.30.103.2 works!
!
R1#ping 172.30.103.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.103.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/36/36 ms
R1#telnet 172.30.103.2
Trying 172.30.103.2 ... Open


User Access Verification

Password:
R2>exit

[Connection to 172.30.103.2 closed by foreign host]
R1#telnet 172.30.102.2
Trying 172.30.102.2 ... Open


User Access Verification

Password:
R2>enable
Password:
!
! Turning on Web server on R2, step 3.
!
R2#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#ip http server
R2(config)#^Z
R2#copy run start
Destination filename [startup-config]?
Building configuration...

!
! Weird but effective test of web server next, step 5
!
```

Example C-4 *Solution to Lab 2 (Continued)*

```
R1#telnet 172.30.103.2 80
Trying 172.30.103.2, 80 ... Open
?HTTP/1.0 400 Bad Request
Date: Mon, 01 Mar 1993 03:03:31 UTC
Content-type: text/html
Expires: Thu, 16 Feb 1989 00:00:00 GMT

<H1>400 Bad Request</H1>



[Connection to 172.30.103.2 closed by foreign host]
R1#
!
! As seen above, after typing the ?, the web connection is broken
! when using this little trick.
!
[Resuming connection 1 to 172.30.102.2 ... ]
[OK]
R2#
R2#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#access-list 101 permit icmp any 172.30.103.0 0.0.0.255
R2(config)#access-list 101 deny tcp any host 172.30.103.2 eq telnet
R2(config)#access-list 101 permit tcp any 172.30.103.0 0.0.0.255 eq telnet
R2(config)#access-list 101 permit TCP any 172.30.103.0 0.0.0.255 eq www
R2(config)#access-list 101 permit tcp any host 172.30.102.2
R2(config)#interface serial 0
R2(config-if)#ip access-group 101 in
!
! Suspending back to R1 w/ Ctrl-Shift-6, x
! Step 8 next. ping should work, Telnet shouldn't, and web should.
!
R1#ping 172.30.103.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.103.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/40/52 ms
R1#telnet 172.30.103.2
Trying 172.30.103.2 ...
% Destination unreachable; gateway or host down

R1#telnet 172.30.103.2 80
Trying 172.30.103.2, 80 ... Open
?HTTP/1.0 400 Bad Request
Date: Mon, 01 Mar 1993 03:08:27 UTC
Content-type: text/html
```

continues

Example C-4 *Solution to Lab 2 (Continued)*

```
Expires: Thu, 16 Feb 1989 00:00:00 GMT

<H1>400 Bad Request</H1>



[Connection to 172.30.103.2 closed by foreign host]
!
! Looks like the access list worked! Go disable it on R2 (step 9)
!
R1#
[Resuming connection 1 to r2 ... ]

R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#int s 0
R2(config-if)#no ip access-group 101 in
R2(config-if)#^Z
R2#
!
! Step 11 next  configure a named access list to do the same.
!

R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#ip access-list ?
  extended    Extended Access List
  log-update  Control access list log updates
  logging     Control access list logging
  standard    Standard Access List

R2(config)#ip access-list extended fred
R2(config-ext-nacl)#permit icmp any 172.30.103.0 0.0.0.255
R2(config-ext-nacl)#deny tcp any host 172.30.103.2 eq telnet
R2(config-ext-nacl)#permit tcp any 172.30.103.0 0.0.0.255 eq www
R2(config-ext-nacl)#permit tcp any host 172.30.102.2 eq telnet
R2(config-ext-nacl)#interface serial 0
R2(config-if)#ip access-group fred in
R2(config-if)#^Z
R2#
!
! Suspending back to R1 to test. ping should work, Telnet should fail,
! Telnet trick to port 80 to test web should work.
!
R1#ping 172.30.103.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.103.2, timeout is 2 seconds:
```

Example C-4 *Solution to Lab 2 (Continued)*

```
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/36/36 ms
R1#telnet 172.30.103.2
Trying 172.30.103.2 ...
% Destination unreachable; gateway or host down

R1#telnet 172.30.103.2 80
Trying 172.30.103.2, 80 ... Open
?HTTP/1.0 400 Bad Request
Date: Mon, 01 Mar 1993 03:10:49 UTC
Content-type: text/html
Expires: Thu, 16 Feb 1989 00:00:00 GMT

<H1>400 Bad Request</H1>



[Connection to 172.30.103.2 closed by foreign host]
R1#
[Resuming connection 1 to r2 ... ]

R2#show access-list
Extended IP access list 101
    permit icmp any 172.30.103.0 0.0.0.255 (10 matches)
    deny tcp any host 172.30.103.2 eq telnet (71 matches)
    permit tcp any 172.30.103.0 0.0.0.255 eq telnet
    permit tcp any 172.30.103.0 0.0.0.255 eq www (12 matches)
    permit tcp any host 172.30.102.2 (181 matches)
Extended IP access list fred
    permit icmp any 172.30.103.0 0.0.0.255 (10 matches)
    deny tcp any host 172.30.103.2 eq telnet (2 matches)
    permit tcp any 172.30.103.0 0.0.0.255 eq www (12 matches)
    permit tcp any host 172.30.102.2 eq telnet (49 matches)
!
! Disabling the named access list next, step 12.
!
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#int s0
R2(config-if)#no ip access-group fred in
R2(config-if)#^Z
R2#copy run start
Destination filename [startup-config]?
Building configuration...

!
! Suspending back to R1, to perform step 13.
!
```

continues

Example C-4 *Solution to Lab 2 (Continued)*

```
R1#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#access-list 101 permit icmp 172.30.103.0 0.0.0.255 any
R1(config)#access-list 101 deny tcp host 172.30.103.2 eq telnet any
R1(config)#access-list 101 permit tcp 172.30.103.0 0.0.0.255 eq telnet any
R1(config)#access-list 101 permit tcp 172.30.103.0 0.0.0.255 eq www any
R1(config)#access-list 101 permit tcp host 172.30.102.2 eq telnet any
R1(config)#interface serial 0
R1(config-if)#ip access-group 101 in
R1(config-if)#^Z
!
! Testing at R1 (step 15). ping should work, Telnet should fail,
! Telnet trick to port 80 to test web should work.
!

R1#ping 172.30.103.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.103.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/35/36 ms
R1#telnet 172.30.103.2
Trying 172.30.103.2 ...
% Connection timed out; remote host not responding

R1#telnet 172.30.103.2 80
Trying 172.30.103.2, 80 ... Open
?HTTP/1.0 400 Bad Request
Date: Mon, 01 Mar 1993 03:14:29 UTC
Content-type: text/html
Expires: Thu, 16 Feb 1989 00:00:00 GMT

<H1>400 Bad Request</H1>



[Connection to 172.30.103.2 closed by foreign host]
!
! Disabling access list on R1, because next labs don't want any access lists. Step 15.
!
R1#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int s 0
R1(config-if)#no ip access-group 101 in
R1(config-if)#^Z
R1#write memory
Building configuration...
```

Example C-4 *Solution to Lab 2 (Continued)*

```
01:56:16: %SYS-5-CONFIG_I: Configured from console by console
[OK]
R1#
```

Lab 3: WAN Configuration

Example C-5 *Solutions to, Lab 3*

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int s 0
R1(config-if)#encapsulation ppp
R1(config-if)#^Z
R1#
00:03:32: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0, changed state to down
00:03:32: %SYS-5-CONFIG_I: Configured from console by console
R1#ping 172.30.102.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.102.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
!
! Previous ping failed, because at step 2, one side of link is PPP,
! the other is HDLC. Line protocol is down in next command.
!
R1#show interface serial 0
Serial0 is up, line protocol is down
  Hardware is HD64570
  Internet address is 172.30.102.1/24
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, loopback not set
  Keepalive set (10 sec)
  LCP Listen
  Closed: IPCP, CDPCP
  Last input 00:00:05, output 00:00:13, output hang never
  Last clearing of "show interface" counters 00:00:31
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/1/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
     Available Bandwidth 1158 kilobits/sec
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     6 packets input, 166 bytes, 0 no buffer
```

continues

Example C-5 *Solutions to, Lab 3 (Continued)*

```
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     1 input errors, 1 CRC, 0 frame, 0 overrun, 0 ignored, 1 abort
     9 packets output, 126 bytes, 0 underruns
 --More
     0 output errors, 0 collisions, 2 interface resets
     0 output buffer failures, 0 output buffers swapped out
     4 carrier transitions
     DCD=up  DSR=up  DTR=up  RTS=up  CTS=up
R1#
!
! Moving to R2 console now. Step 3 next.
!
R2>
R2>enable
Password:
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#int s 0
R2(config-if)#encapsulation ppp
R2(config-if)#^Z
R2#
!
! Moving to R1 console now. Step 3 testing next.
!
R1#
R1#ping 172.30.102.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.102.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/36/36 ms
R1#show interface serial 0
Serial0 is up, line protocol is up
  Hardware is HD64570
  Internet address is 172.30.102.1/24
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, loopback not set
  Keepalive set (10 sec)
  LCP Open
  Open: IPCP, CDPCP
  Last input 00:00:05, output 00:00:05, output hang never
  Last clearing of "show interface" counters 00:01:13
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/1/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
```

Example C-5 *Solutions to, Lab 3 (Continued)*

```
     Available Bandwidth 1158 kilobits/sec
  5 minute input rate 0 bits/sec, 1 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     24 packets input, 1727 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     1 input errors, 1 CRC, 0 frame, 0 overrun, 0 ignored, 1 abort
     29 packets output, 1960 bytes, 0 underruns
 --More
     0 output errors, 0 collisions, 2 interface resets
     0 output buffer failures, 0 output buffers swapped out
     10 carrier transitions
     DCD=up  DSR=up  DTR=up  RTS=up  CTS=up
R1#debug ppp negotiation
PPP protocol negotiation debugging is on
R1#R2

Trying R2 (172.30.102.2)... Open


User Access Verification

Password:
R2>

!
! Suspending Telnet for now.
!
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface serial 0
R1(config-if)#shutdown
00:05:39: Se0 IPCP: State is Closed
00:05:39: Se0 CDPCP: State is Closed
00:05:39: Se0 PPP: Phase is TERMINATING [0 sess, 1 load]
00:05:39: Se0 LCP: State is Closed
00:05:39: Se0 PPP: Phase is DOWN [0 sess, 1 load]
00:05:39: Se0 IPCP: Remove route to 172.30.102.2
00:05:40: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0, changed state to down
R1(config-if)#no shutdown
R1(config-if)#^Z
R1#
00:05:48: %SYS-5-CONFIG_I: Configured from console by console
00:05:48: %LINK-3-UPDOWN: Interface Serial0, changed state to up
00:05:48: Se0 PPP: Treating connection as a dedicated line
00:05:48: Se0 PPP: Phase is ESTABLISHING, Active Open [0 sess, 1 load]
00:05:48: Se0 LCP: O CONFREQ [Closed] id 15 len 10
00:05:48: Se0 LCP:    MagicNumber 0x00118F3A (0x050600118F3A)
00:05:48: Se0 LCP: I CONFREQ [REQsent] id 2 len 10
```

continues

Example C-5 *Solutions to, Lab 3 (Continued)*

```
00:05:48: Se0 LCP:    MagicNumber 0x0227C7F9 (0x05060227C7F9)
00:05:48: Se0 LCP: O CONFACK [REQsent] id 2 len 10
00:05:48: Se0 LCP:    MagicNumber 0x0227C7F9 (0x05060227C7F9)
00:05:48: Se0 LCP: I CONFACK [ACKsent] id 15 len 10
00:05:48: Se0 LCP:    MagicNumber 0x00118F3A (0x050600118F3A)
00:05:48: Se0 LCP: State is Open
00:05:48: Se0 PPP: Phase is UP [0 sess, 1 load]
00:05:48: Se0 IPCP: O CONFREQ [Closed] id 2 len 10
00:05:48: Se0 IPCP:    Address 172.30.102.1 (0x0306AC1E6601)
00:05:48: Se0 CDPCP: O CONFREQ [Closed] id 2 len 4
00:05:48: Se0 IPCP: I CONFREQ [REQsent] id 2 len 10
00:05:48: Se0 IPCP:    Address 172.30.102.2 (0x0306AC1E6602)
00:05:48: Se0 IPCP: O CONFACK [REQsent] id 2 len 10
00:05:48: Se0 IPCP:    Address 172.30.102.2 (0x0306AC1E6602)
00:05:48: Se0 CDPCP: I CONFREQ [REQsent] id 2 len 4
00:05:48: Se0 CDPCP: O CONFACK [REQsent] id 2 len 4
00:05:48: Se0 IPCP: I CONFACK [ACKsent] id 2 len 10
00:05:48: Se0 IPCP:    Address 172.30.102.1 (0x0306AC1E6601)
00:05:48: Se0 IPCP: State is Open
00:05:48: Se0 CDPCP: I CONFACK [ACKsent] id 2 len 4
00:05:48: Se0 CDPCP: State is Open
00:05:48: Se0 IPCP: Install route to 172.30.102.2
00:05:49: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0, changed state to up
!
! Previous lines of output show details of step 7.
! Step 8 next.
!
R1#no debug all
!
! R3 has already been configured. Copy the configuration from earlier in
! this chapter to get R3 configured as a Frame Relay switch.
!

R1#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface serial 0
R1(config-if)#shutdown
R1(config-if)#interface serial 1
R1(config-if)#encapsulation frame-relay
R1(config-if)#ip address 172.30.104.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#^Z
00:07:34: %LINK-3-UPDOWN: Interface Serial1, changed state to up
R1#
00:07:40: %SYS-5-CONFIG_I: Configured from console by console
R1#copy run start
Destination filename [startup-config]?
Building configuration...
```

Example C-5 *Solutions to, Lab 3 (Continued)*

```
!
! Moved console cable to R2, still in step 8-9
!
R2#
R2#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface serial 0
R2(config-if)#shutdown
R2(config-if)#interface serial 1
R2(config-if)#encapsulation frame-relay
R2(config-if)#ip address 172.30.104.2 255.255.255.0
R2(config-if)#no shutdown
R2(config-if)#^Z
R2#
09:51:47: %SYS-5-CONFIG_I: Configured from console by console

09:51:49: %LINK-3-UPDOWN: Interface Serial1, changed state to up

R2#write memory

Building configuration...

!
! Moving console cable back to R1.  Step 12 next.
!
R1#show interface serial 1
Serial1 is up, line protocol is up
  Hardware is HD64570
  Internet address is 172.30.104.1/24
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation FRAME-RELAY, loopback not set
  Keepalive set (10 sec)
  LMI enq sent  8, LMI stat recvd 10, LMI upd recvd 0, DTE LMI up
  LMI enq recvd 0, LMI stat sent  0, LMI upd sent  0
  LMI DLCI 1023  LMI type is CISCO  frame relay DTE
  Broadcast queue 0/64, broadcasts sent/dropped 1/0, interface broadcasts 0
  Last input 00:00:01, output 00:00:01, output hang never
  Last clearing of "show interface" counters 00:01:39
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: weighted fair
  Output queue: 0/1000/64/0 (size/max total/threshold/drops)
     Conversations  0/1/256 (active/max active/max total)
     Reserved Conversations 0/0 (allocated/max allocated)
     Available Bandwidth 1158 kilobits/sec
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     12 packets input, 238 bytes, 0 no buffer
```

continues

Example C-5 *Solutions to, Lab 3 (Continued)*

```
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
 --More
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     16 packets output, 252 bytes, 0 underruns
     0 output errors, 0 collisions, 1 interface resets
     0 output buffer failures, 0 output buffers swapped out
     2 carrier transitions
     DCD=up  DSR=up  DTR=up  RTS=up  CTS=up
R1#ping 172.30.104.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.104.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/135/404 ms
R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 3 subnets
I       172.30.103.0 [100/8576] via 172.30.104.2, 00:00:16, Serial1
C       172.30.101.0 is directly connected, Ethernet0
C       172.30.104.0 is directly connected, Serial1
R1#ping 172.30.103.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.103.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/68/72 ms
R1#r2
Trying R2 (172.30.102.2)...
% Destination unreachable; gateway or host down
Trying R2 (172.30.103.2)... Open


User Access Verification

Password:
R2>ena
Password:

!
```

Example C-5 *Solutions to, Lab 3 (Continued)*

```
! Telnetted to R2 to test from there. Step 12.
!
R2#ping 172.30.104.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.104.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 72/152/440 ms
R2#ping 172.30.101.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.101.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 72/79/100 ms
R2#
!
! Suspended back to R1. Step 13 next.
!
R1#show frame-relay map
Serial1 (up): ip 172.30.104.2 dlci 202(0xC9,0x3090), dynamic,
              broadcast,, status defined, active
R1#show frame-relay pvc

PVC Statistics for interface Serial1 (Frame Relay DTE)

              Active     Inactive      Deleted       Static
  Local          1            0            0            0
  Switched       0            0            0            0
  Unused         0            0            0            0

DLCI = 201, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial1

  input pkts 109           output pkts 120          in bytes 6990
  out bytes 6936           dropped pkts 0           in FECN pkts 0
  in BECN pkts 0           out FECN pkts 0          out BECN pkts 0
  in DE pkts 0             out DE pkts 0
  out bcast pkts 3         out bcast bytes 190
  pvc create time 00:03:09, last time pvc status changed 00:01:59
!
! Step 14 next.
!
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface serial 1
R1(config-if)#no frame-relay inverse-arp
R1(config-if)#^Z
R1#r2
Trying R2 (172.30.102.2)...
```

continues

Example C-5 *Solutions to, Lab 3 (Continued)*

```
% Destination unreachable; gateway or host down
Trying R2 (172.30.103.2)... Open

00:11:25: %SYS-5-CONFIG_I: Configured from console by console

User Access Verification

Password:
R2>ena
Password:
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface serial 1
R2(config-if)#no frame-relay inverse-arp
R2(config-if)#^Z
R2#
R1#clear frame-relay inarp
R2#clear frame-relay inarp

R1#show frame-relay map
!
! After clearing inverse ARP cache, the cache is empty, because Inverse
! ARP is disabled now.
!
R1#ping 172.30.104.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.104.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
R1#
!
! Need static maps, now that Inverse ARP is disabled. Step 15.
!
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface serial 1
R1(config-if)#frame-relay map ip 172.30.104.2 202 broadcast
R1(config-if)#^Z
R1#
00:14:03: %SYS-5-CONFIG_I: Configured from console by console
00:14:21: %LINK-3-UPDOWN: Interface Serial1, changed state to down
00:14:22: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1, changed state to down
00:14:46: %LINK-3-UPDOWN: Interface Serial1, changed state to up
00:14:47: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial1, changed state to up
R1#show frame-relay map
Serial1 (up): ip 172.30.104.2 dlci 202(0xCA,0x30A0), static,
              CISCO, status defined, active
```

Example C-5 *Solutions to, Lab 3 (Continued)*

```
R1#show frame-relay pvc

PVC Statistics for interface Serial1 (Frame Relay DTE)

             Active     Inactive      Deleted       Static
  Local          1            0            0            0
  Switched       0            0            0            0
  Unused         0            0            0            0

DLCI = 202, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial1

  input pkts 0             output pkts 2            in bytes 0
  out bytes 90             dropped pkts 0           in FECN pkts 0
  in BECN pkts 0           out FECN pkts 0          out BECN pkts 0
  in DE pkts 0             out DE pkts 0
  out bcast pkts 0         out bcast bytes 0
  pvc create time 00:01:15, last time pvc status changed 00:00:20
!
! Moving console cable back to R2, so map can be added. Step 15.
!
R2#
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface serial 1
R2(config-if)#frame-relay map ip 172.30.104.1 201 broadcast
R2(config-if)#^Z
R2#ping 172.30.104.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.104.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/68/68 ms
!
! Moved console cable back to R1.
!
R1#
R1#ping 172.30.104.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.104.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/139/424 ms
R1#show frame-relay map
Serial1 (up): ip 172.30.104.2 dlci 202(0xCA,0x30A0), static,
              CISCO, status defined, active
R1#
!
! Step 16 comes next. Get rid of Frame Relay configuration by changing
```

continues

Example C-5 *Solutions to, Lab 3 (Continued)*

```
! the encapsulation to HDLC.
!
R1#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface serial 1
R1(config-if)#encap hdlc
R1(config-if)#^Z
R1#show runing-config
Building configuration...

Current configuration : 1045 bytes
!
version 12.2
no service single-slot-reload-enable
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname R1
!
logging rate-limit console 10 except errors
enable secret 5 $1$dWq2$3HqStateLGu3QYhjwTUkw0
enable password fred
!
ip subnet-zero
no ip finger
ip host R2 172.30.102.2 172.30.103.2
!
no ip dhcp-client network-discovery
!
!
!
 --More
interface Ethernet0
 ip address 172.30.101.1 255.255.255.0
!
interface Serial0
 ip address 172.30.102.1 255.255.255.0
 encapsulation ppp
 shutdown
!
interface Serial1
 ip address 172.30.104.1 255.255.255.0
!
!
! Deleted rest of config for brevity's sake.
!
R1#config t
Enter configuration commands, one per line.  End with CNTL/Z.
```

Example C-5 *Solutions to, Lab 3 (Continued)*

```
R1(config)#interface serial 1
R1(config-if)#no ip address
R1(config-if)#encapsulation frame-relay
R1(config-if)#interface serial 1.1 point-to-point
R1(config-subif)#ip address 172.30.104.1 255.255.255.0
R2(config-subif)#frame-relay interface-dlci 202
R1(config-subif)#^Z
R1#
!
! Moved cable to R2 console, doing step 16.
! Same trick  change encapsulation to HDLC, all the Frame Relay
! configuration gets deleted.
!
R2#config t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#int s 1
R2(config-if)#encap hdlc
R2(config-if)#no ip address
R2(config-if)#^Z
R2#
R2#show running-config
Building configuration...

Current configuration : 1059 bytes
!
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname R2
!
enable secret 5 $1$/aKj$cCiKYkzHbrHRfkPi9VBAw.
enable password fred
!
ip subnet-zero
!
!
!
!
interface Ethernet0
 ip address 172.30.103.2 255.255.255.0
!
interface Serial0
 ip address 172.30.102.2 255.255.255.0
 encapsulation ppp
shutdown
 no fair-queue
```

continues

Example C-5 *Solutions to, Lab 3 (Continued)*

```
 clockrate 56000
!
interface Serial1
 no ip address
!
! Rest of config not shown to save space.
!
R2#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface serial 1
R2(config-if)#encapsulation frame-relay
R2(config-if)#interface serial 1.1 point-to-point
R2(config-subif)#ip address 172.30.104.2 255.255.255.0
R2(config-subif)#frame-relay interface-dlci 201
R2(config-subif)#^Z
R2#
R2#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]

!
! Moved back to R1 console. Step  16, testing now…
!
R1#ping 172.30.104.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.104.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/140/428 ms
R1#write memory
Building configuration...
[OK]
R1#show frame-relay pvc

PVC Statistics for interface Serial1 (Frame Relay DTE)

             Active     Inactive      Deleted       Static
  Local          1            0            0            0
  Switched       0            0            0            0
  Unused         0            0            0            0

DLCI = 202, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial1.1

  input pkts 6             output pkts 8            in bytes 570
  out bytes 897            dropped pkts 0           in FECN pkts 0
  in BECN pkts 0           out FECN pkts 0          out BECN pkts 0
  in DE pkts 0             out DE pkts 0
```

Example C-5 *Solutions to, Lab 3 (Continued)*

```
  out bcast pkts 3         out bcast bytes 377
  pvc create time 00:00:34, last time pvc status changed 00:00:34
!
! At step 17, map information is static, but it's the same info
! as when inverse ARP was used.
!
R1#show frame-relay map
Serial1.1 (up): point-to-point dlci, dlci 202(0xCA,0x30A0), broadcast
          status defined, active
R1#ping 172.30.103.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.103.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/68/68 ms
R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.30.0.0/24 is subnetted, 3 subnets
I       172.30.103.0 [100/8576] via 172.30.104.2, 00:00:16, Serial1.1
C       172.30.101.0 is directly connected, Ethernet0
C       172.30.104.0 is directly connected, Serial1.1
!
! About to do step 18.
!
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#int serial 1
R1(config-if)#encapsulation frame-relay ietf
R1(config-if)#
R1#
R1#ping 172.30.104.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.104.2, timeout is 2 seconds:
....
!
! Step 19 next. After R2 changes to IETF, ping works again.
!
```

continues

Example C-5 *Solutions to, Lab 3 (Continued)*

```
R2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R2(config)#interface serial 1
R2(config-if)#encapsulation frame-relay ietf
R2(config-if)#^Z
R2#
10:13:02: %SYS-5-CONFIG_I: Configured from console by consoleÿ
R1#ping 172.30.104.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.30.104.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/86/160 ms
!
! Step 20 next…
!
R1#show frame-relay lmi

LMI Statistics for interface Serial1 (Frame Relay DTE) LMI TYPE = CISCO   Invalid Unnumbered
info 0     Invalid Prot Disc 0
   Invalid dummy Call Ref 0      Invalid Msg Type 0
   Invalid Status Message 0      Invalid Lock Shift 0
   Invalid Information ID 0      Invalid Report IE Len 0
   Invalid Report Request 0      Invalid Keep IE Len 0
   Num Status Enq. Sent 67       Num Status msgs Rcvd 68
   Num Update Status Rcvd 0      Num Status Timeouts 0
!
! Reconnect to R2 via suspended Telnet.
!
R2#show frame-relay lmi

LMI Statistics for interface Serial1 (Frame Relay DTE) LMI TYPE = CISCO
   Invalid Unnumbered info 0     Invalid Prot Disc 0
   Invalid dummy Call Ref 0      Invalid Msg Type 0
   Invalid Status Message 0      Invalid Lock Shift 0
   Invalid Information ID 0      Invalid Report IE Len 0
   Invalid Report Request 0      Invalid Keep IE Len 0
   Num Status Enq. Sent 60       Num Status msgs Rcvd 61
   Num Update Status Rcvd 0      Num Status Timeouts 0
R2#
```